HubSpot

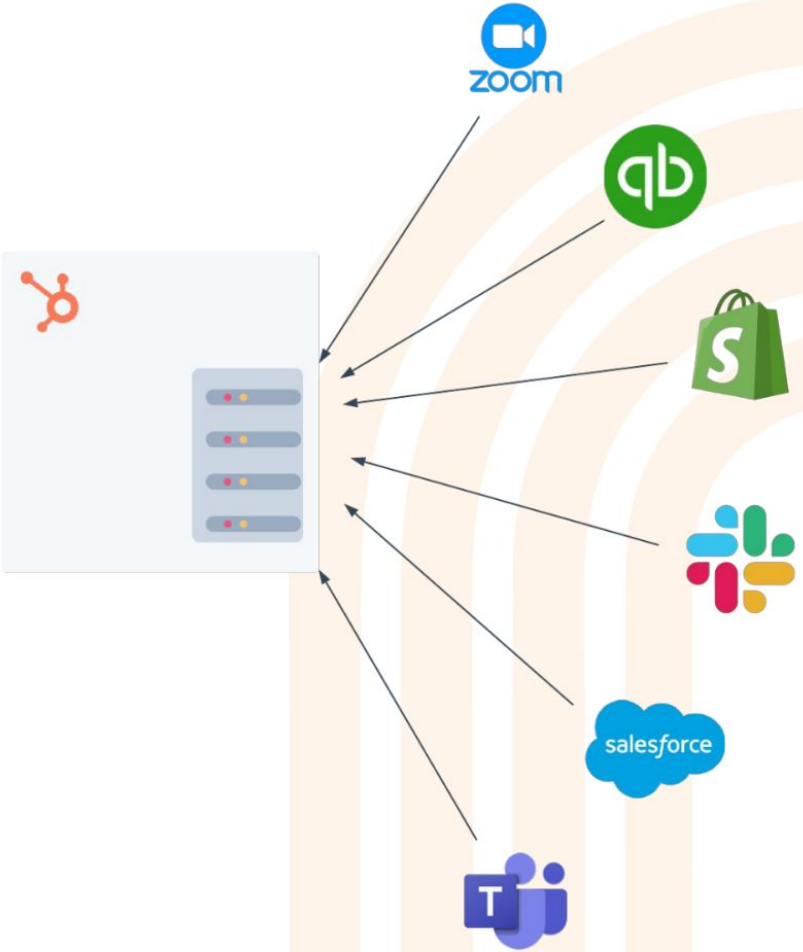
Integration Reliability

by Amanda Bridge

HubSpot

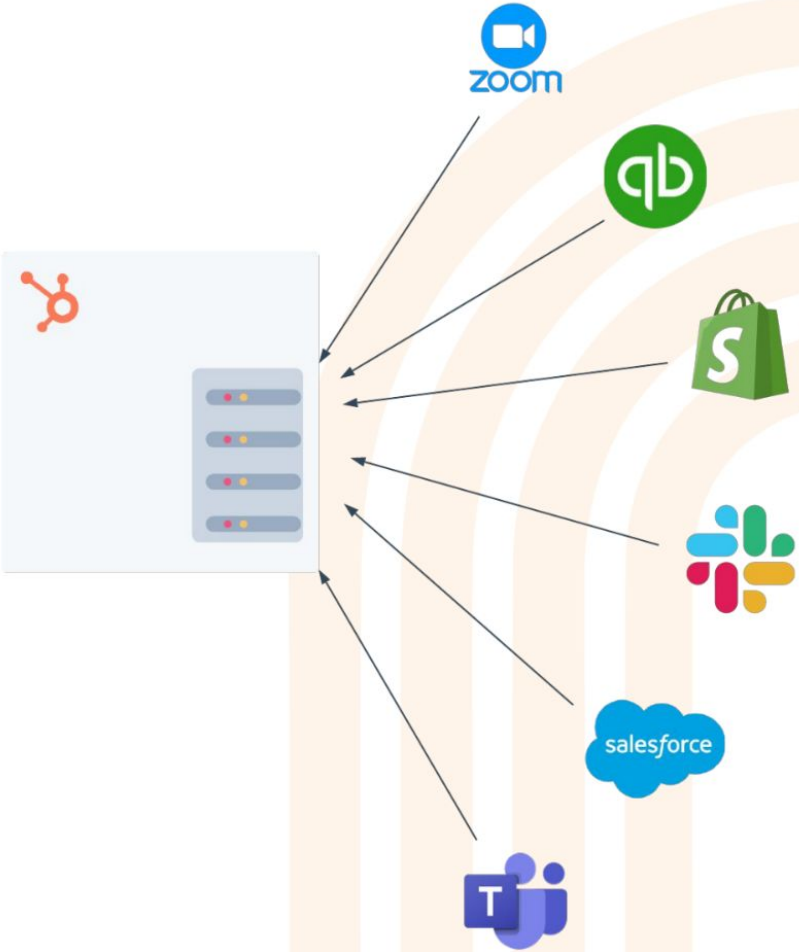
BACKGROUND

l and an engineering lead led an initiative to improve our front end responses when a 3rd party outage affected the uptime of one of our strategic integrations.



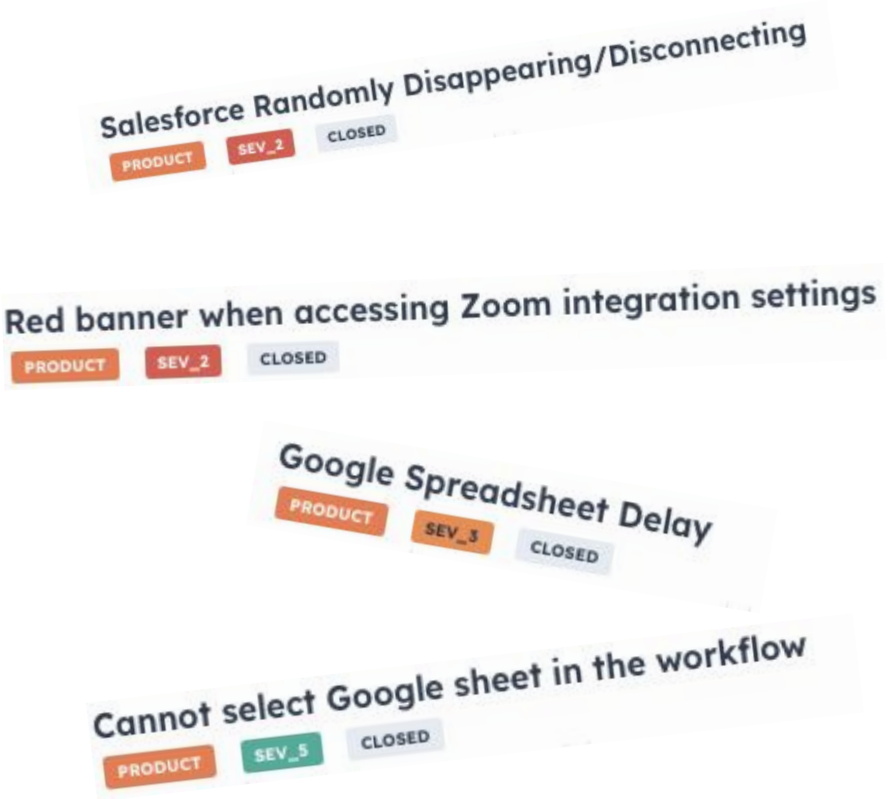
BACKGROUND

80% of pro accounts and **60% of free accounts** have more than 10 integrations installed, and so a lot of our user questions include references to 3rd parties when issues arise.



BACKGROUND

50% of Integration related critsits in 2023 were due to 3rd party outage and breaking changes



Task

TASK

Objectives

1. **Audit all API calls** of our strategic integrations, and identify the most impactful ones to our UX when a 3rd party outage or breaking change affects it.

2. **Audit current UX responses** when each of the API calls is affected.

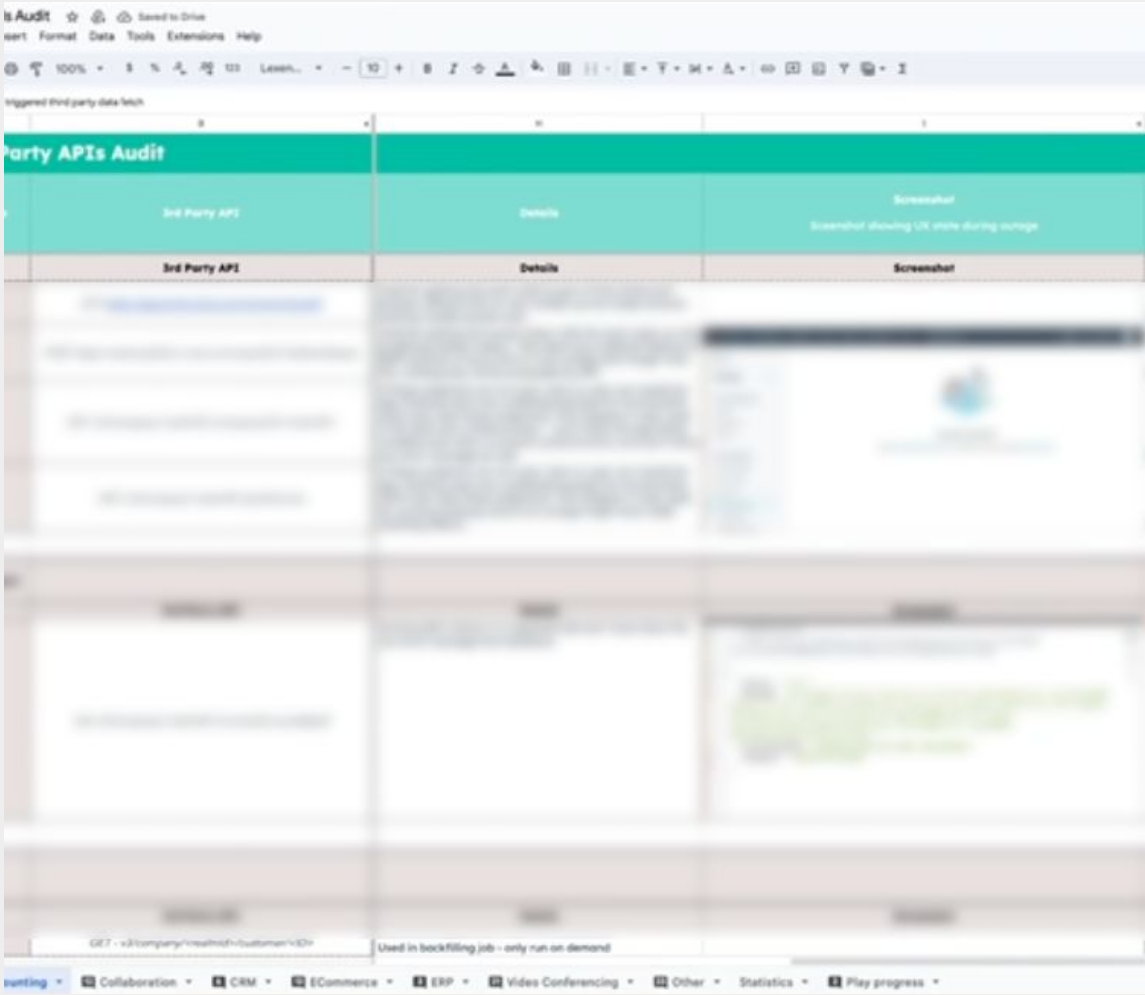
3. Build UX and engineering **best practices** to provide responses for at least 75% of API calls.

4. **Implement changes** based on best practices.

Action

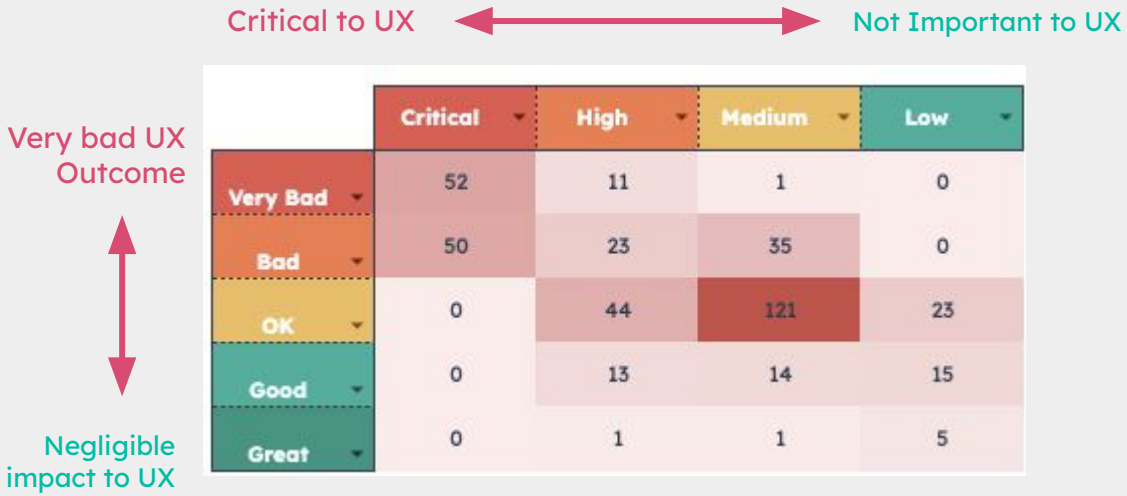
ACTION

API Audit



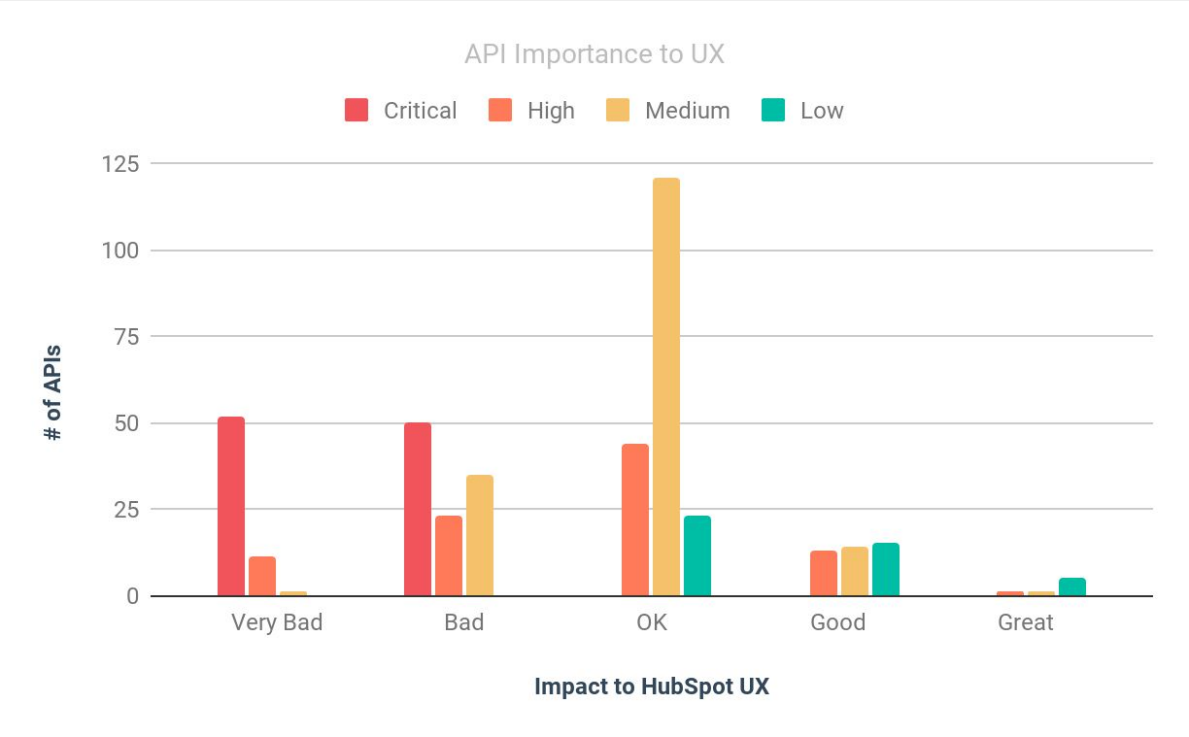
ACTION

API Audit



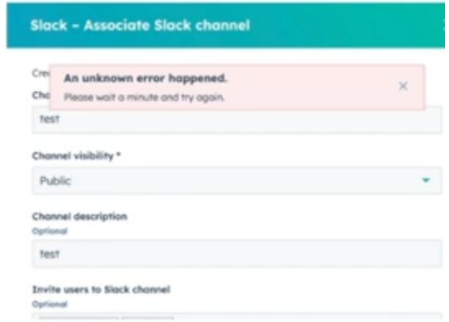
ACTION

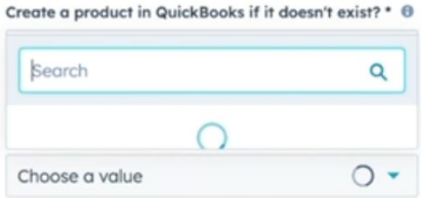
API Audit



BACKGROUND

Current UX responses

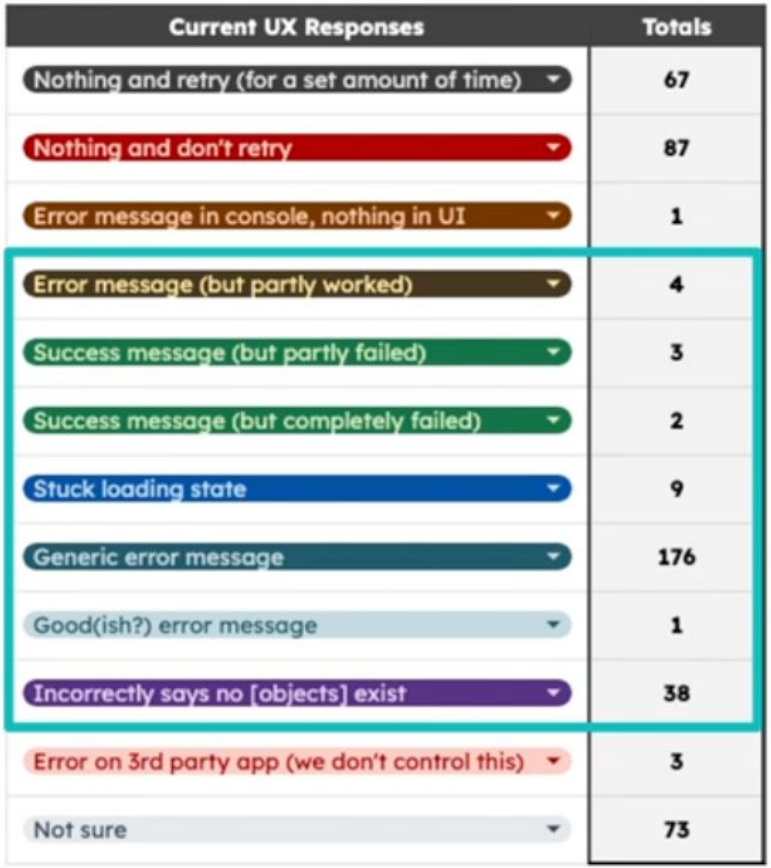




ACTION

UX Audit

50% of the time we don't show users any feedback when an API fails.



Current UX Responses	Totals
Nothing and retry (for a set amount of time)	67
Nothing and don't retry	87
Error message in console, nothing in UI	1
Error message (but partly worked)	4
Success message (but partly failed)	3
Success message (but completely failed)	2
Stuck loading state	9
Generic error message	176
Good(ish?) error message	1
Incorrectly says no [objects] exist	38
Error on 3rd party app (we don't control this)	3
Not sure	73

ACTION

UX Audit

10% of the time our feedback is actually incorrect or false.

Current UX Responses	Totals
Nothing and retry (for a set amount of time)	67
Nothing and don't retry	87
Error message in console, nothing in UI	1
Error message (but partly worked)	4
Success message (but partly failed)	3
Success message (but completely failed)	2
Stuck loading state	9
Generic error message	176
Good(ish?) error message	1
Incorrectly says no [objects] exist	38
Error on 3rd party app (we don't control this)	3
Not sure	73

ACTION

UX Audit

40% of the time, our explanations are either hard to find or not helpful.

Current UX Responses	Totals
Nothing and retry (for a set amount of time)	67
Nothing and don't retry	87
Error message in console, nothing in UI	1
Error message (but partly worked)	4
Success message (but partly failed)	3
Success message (but completely failed)	2
Stuck loading state	9
Generic error message	176
Good(ish?) error message	1
Incorrectly says no [objects] exist	38
Error on 3rd party app (we don't control this)	3
Not sure	73

ACTION

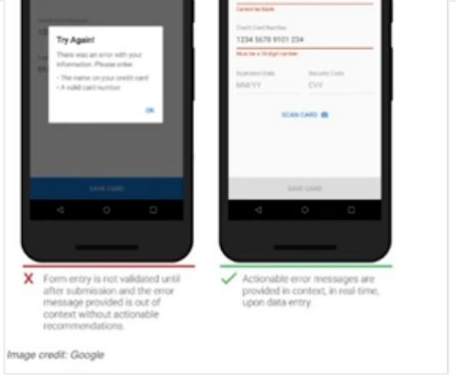
UX Audit

33% of the time, the user isn't given any indication in the UI that the operation has failed.

Current UX Responses	Totals
Nothing and retry (for a set amount of time)	67
Nothing and don't retry	87
Error message in console, nothing in UI	1
Error message (but partly worked)	4
Success message (but partly failed)	3
Success message (but completely failed)	2
Stuck loading state	9
Generic error message	176
Good(ish?) error message	1
Incorrectly says no [objects] exist	38
Error on 3rd party app (we don't control this)	3
Not sure	73

ACTION

Market research



ACTION

Build foundational best practices

Nielsen Norman Group Error Best Practices:

- Design effective error messages by ensuring they are highly **visible**, provide **constructive communication**, and **respect** user effort.

HubSpot Error Message Guidelines:

Don't

- Don't use technical or HubSpot jargon.
- Don't use frivolous language like 'oops', 'uh-oh' and 'it's us, not you'.
- Don't use exclamation marks.
- Don't say sorry or take up space with too many pleasantries.

Do

- Be clear, direct, brief, and helpful.

ACTION

Build foundational best practices

Differences between errors related to 3rd party outages and general errors:

- We **don't control the solution** or its timeline
- Instead of something simply not working, **data may be outdated and incorrect**
- In some cases, **users might not notice something went wrong** without an error
- **Errors are never in response to an incorrect user input** (i.e. even if the user did everything right, the error would still happen)
- Sometimes, these **errors are temporary and will be fixed without any action** from the user.

ACTION

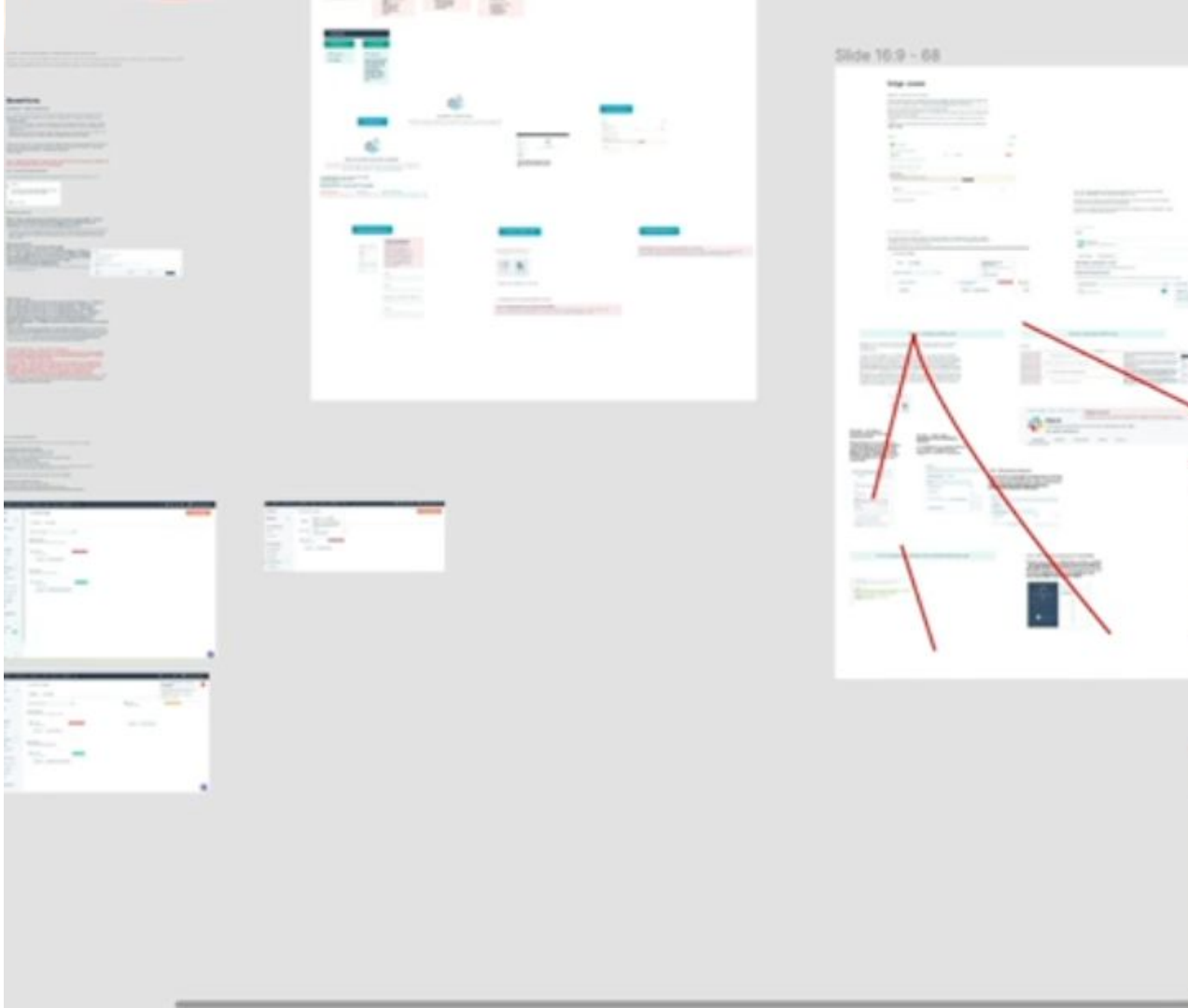
Build foundational best practices

HubSpot best practices for errors relating to third-party outages:

- **Be contextual**. Show the error where the user is working.
- **Be timely.** When possible, warn the user *before* they attempt an action that is temporarily affected by an outage.
- **Take the blame**. Do not blame the 3rd party application, and do not reference the outage in case they haven't publicized it yet. Never blame the user.
- **Be human.** Tell people what happened in the most human way possible. Do not show the user a coded error message.
- **Be actionable.** Although there is often no action needed from the user to resolve the error, always provide the user with a next step.

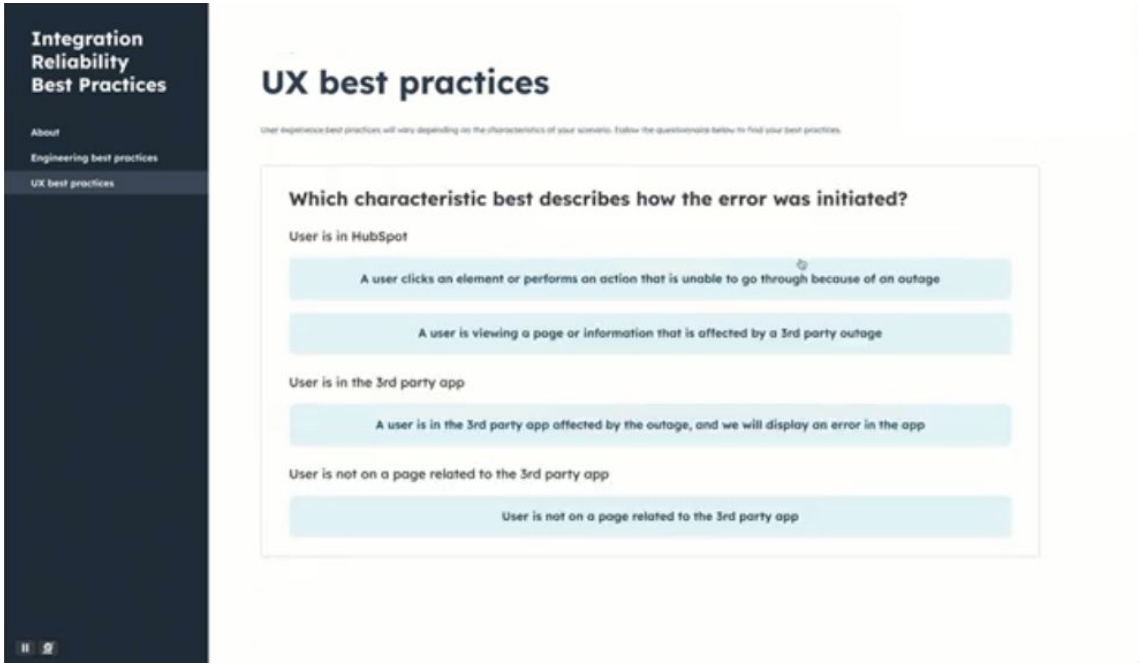
ACTION

Analyze and categorize API UX responses



ACTION

Analyze and categorize API UX responses

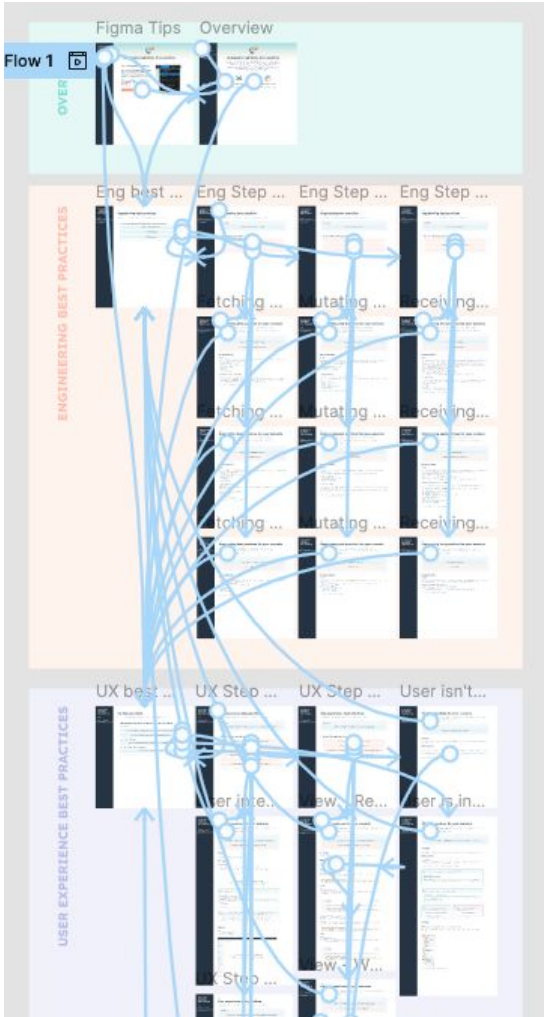


ACTION

Created best practices in Figma that solved for 86% of all API calls

ACTION

Created best practices in Figma that solved for 86% of all API calls



Guidelines walkthrough

SI Reliability Best Practices | SI Reliability Best Practices - ⭐

Integration Reliability Best Practices

- About
- Engineering best practices
- UX best practices

Integration reliability best practices

HubSpot's integrations allow users to connect data between HubSpot and 3rd party applications. However, when a 3rd party app has an outage or breaking change that affects HubSpot, it impacts our users' ability to perform actions and see up-to-date information.

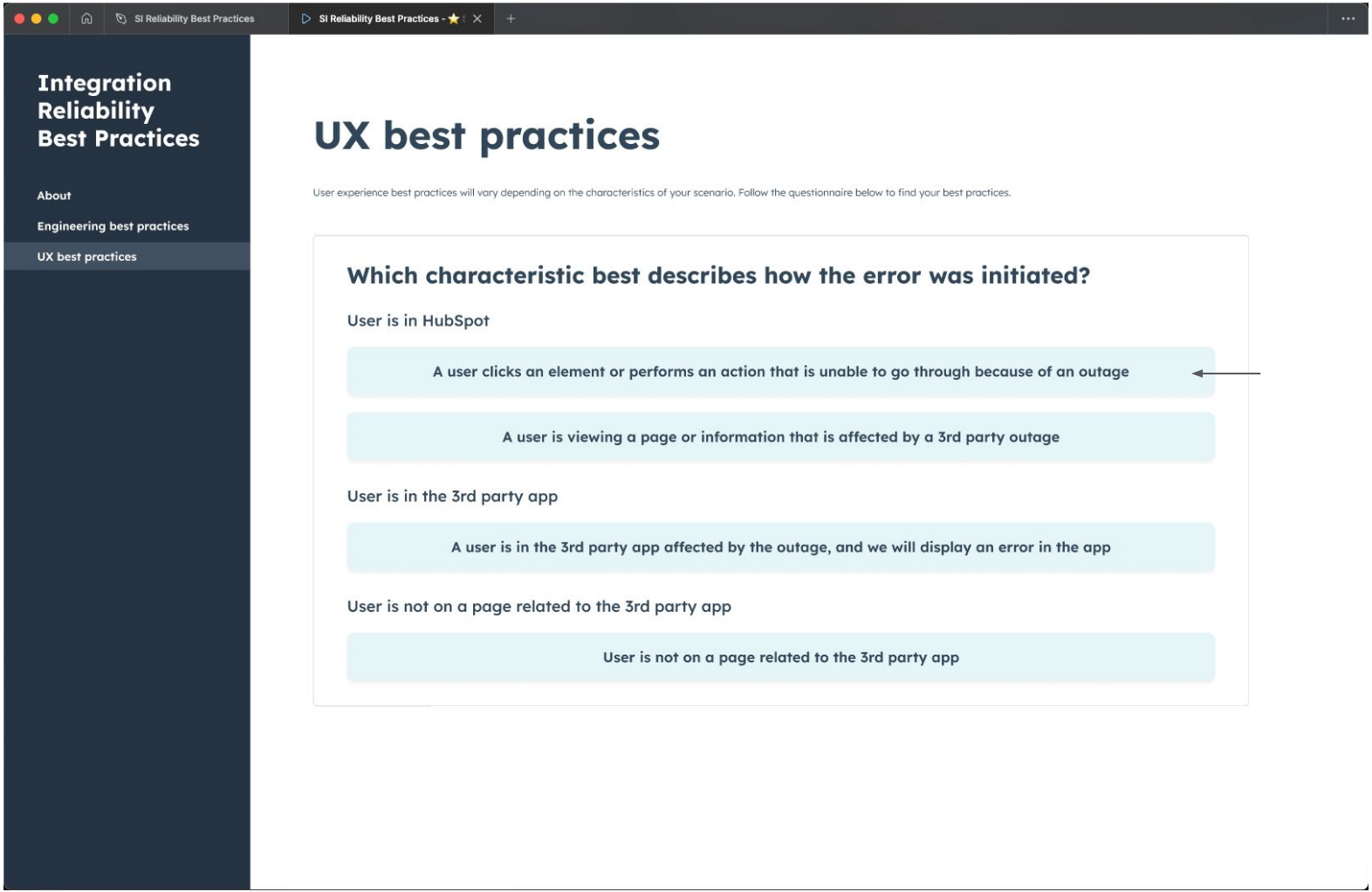
The following best practices inform engineering and user experience design responses to common use cases we see during 3rd party outages.

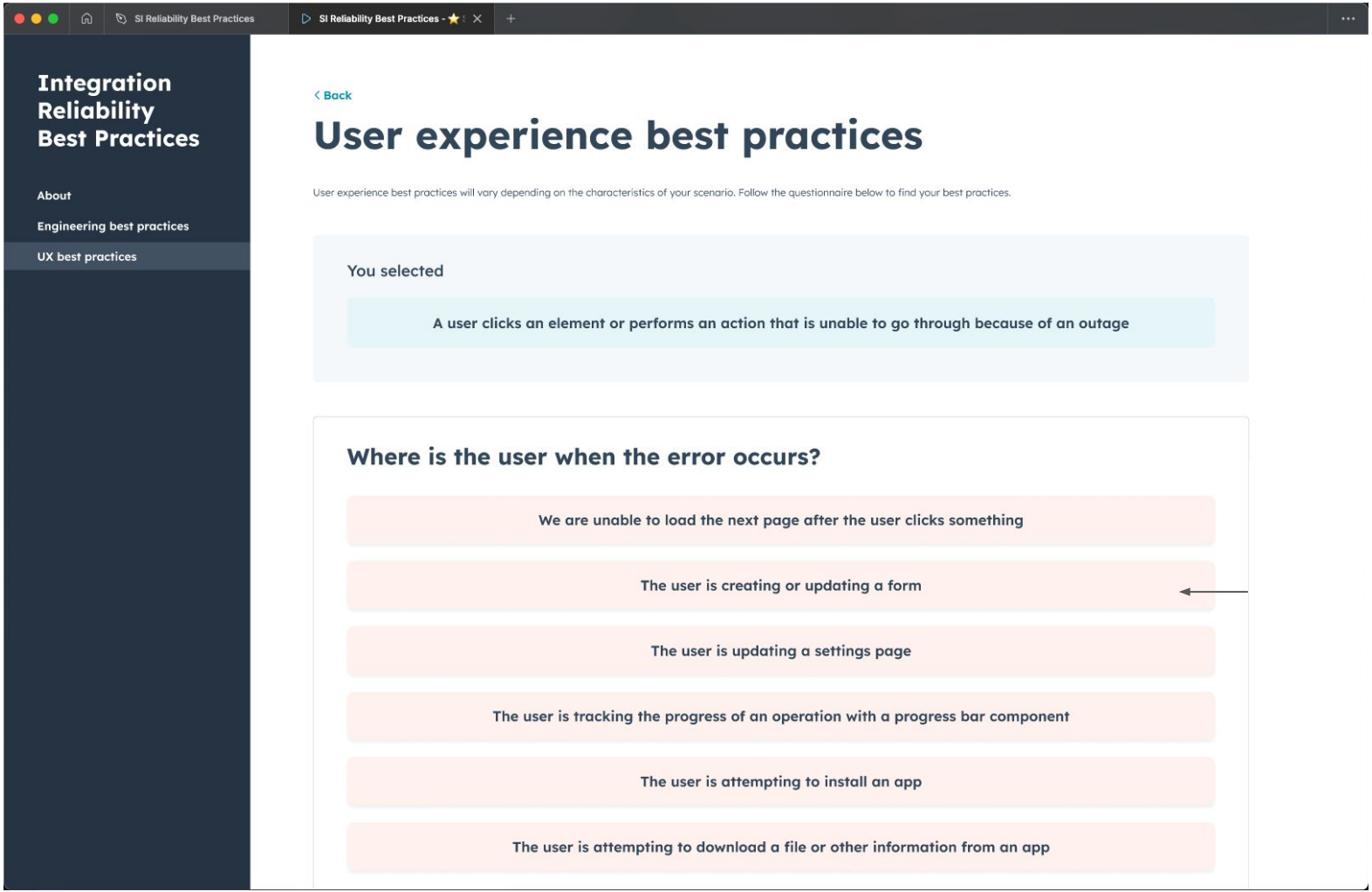
Engineering best practices

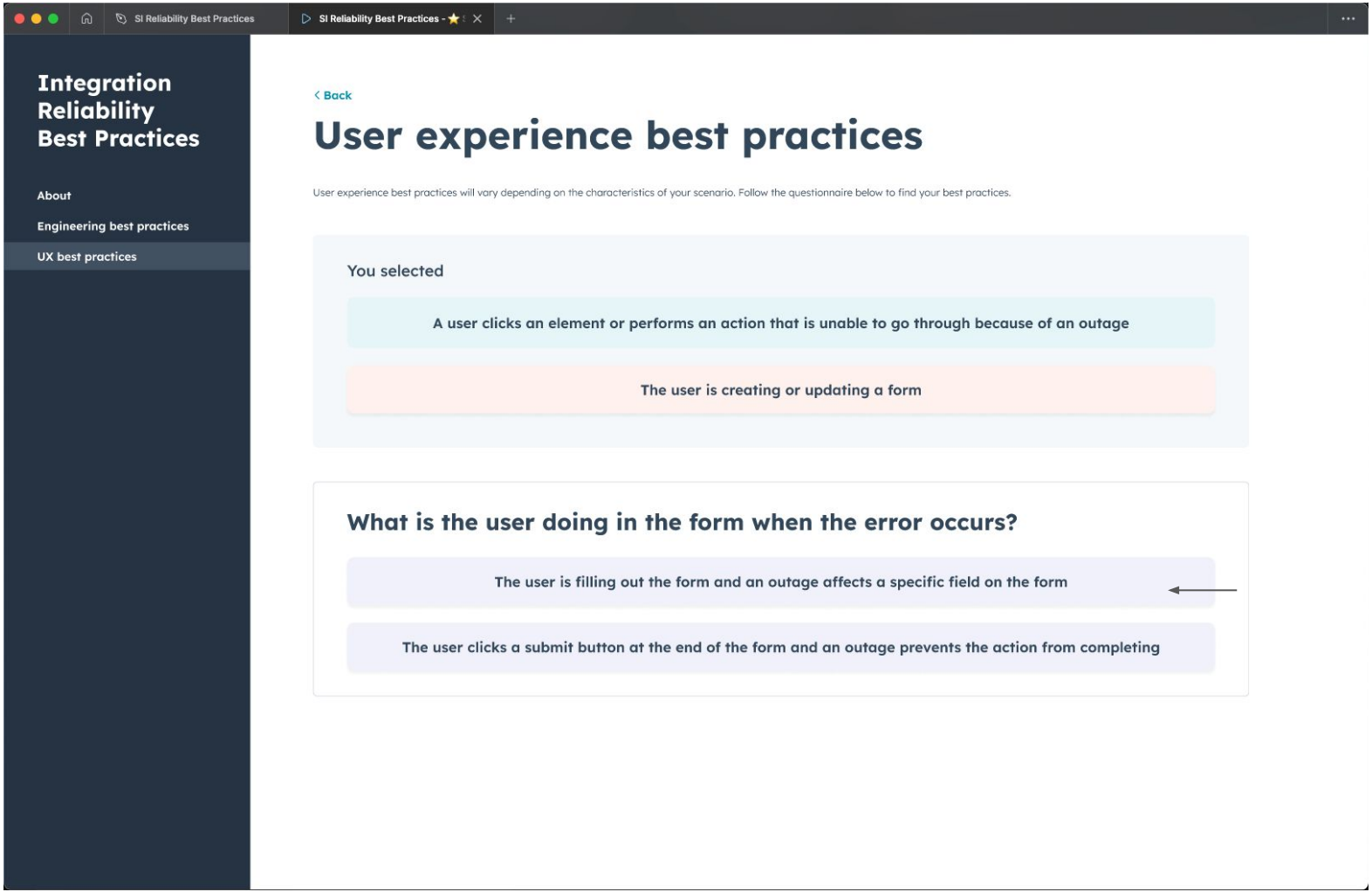
Guidelines for backend engineers to help mitigate risk in the event of a 3rd party outage related to your integration.

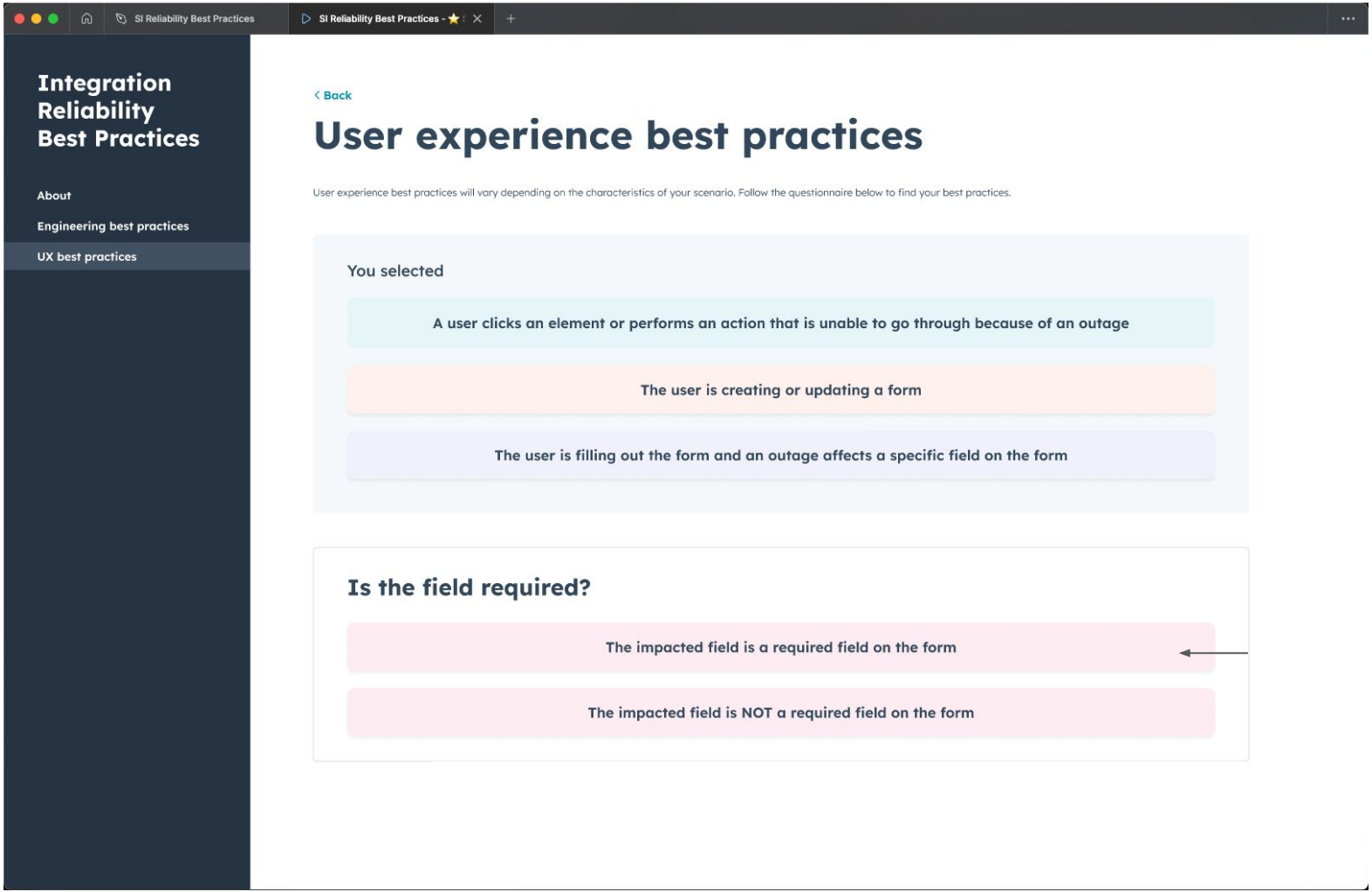
User experience best practices

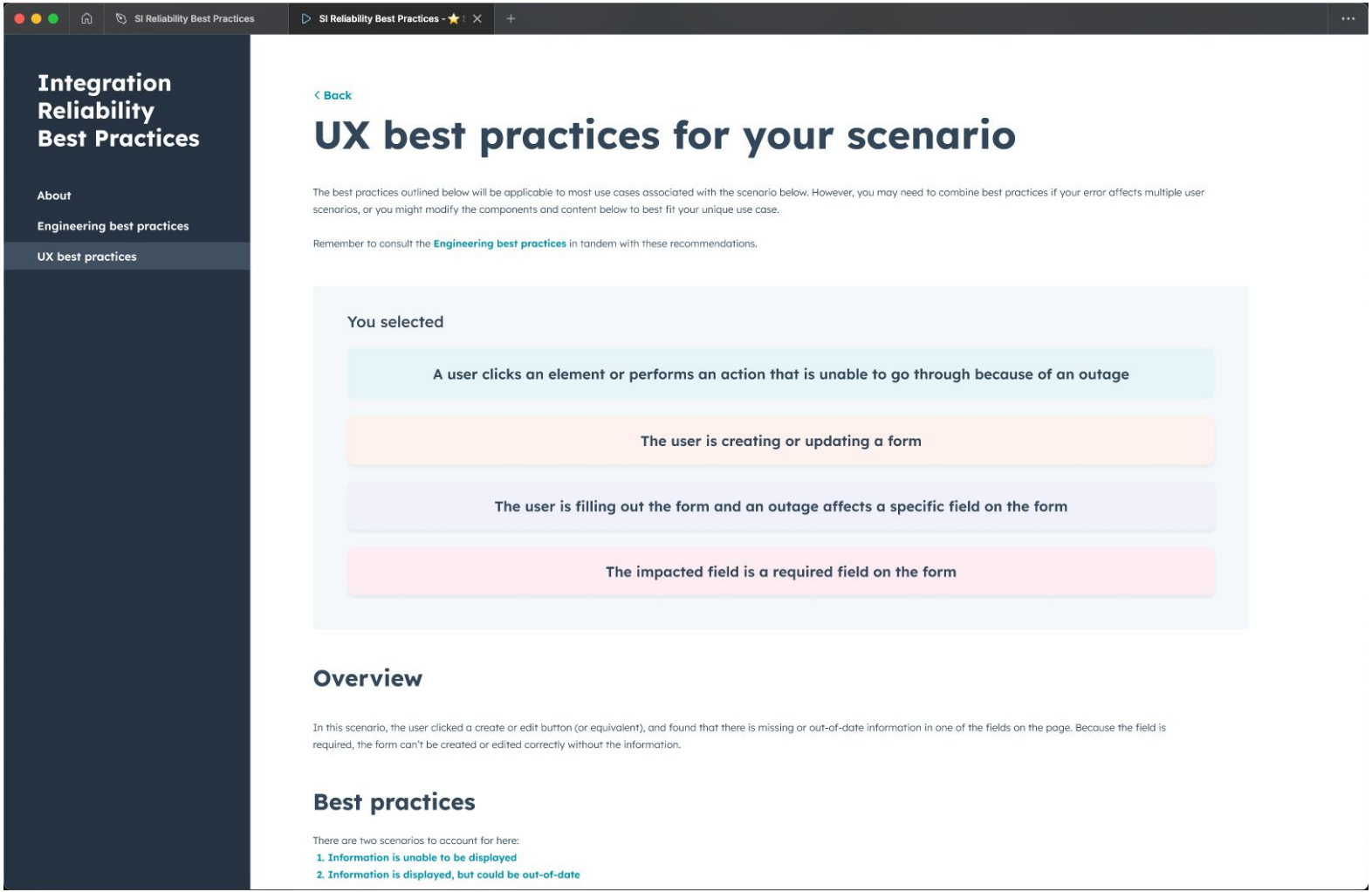
Guidelines for product designers to respond to 3rd party outages that affect our user's experience in HubSpot.


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
UX best practices
User experience best practices will vary depending on the characteristics of your scenario. Follow the questionnaire below to find your best practices.
Which characteristic best describes how the error was initiated?
User is in HubSpot
A user clicks an element or performs an action that is unable to go through because of an outage
A user is viewing a page or information that is affected by a 3rd party outage
User is in the 3rd party app
A user is in the 3rd party app affected by the outage, and we will display an error in the app
User is not on a page related to the 3rd party app
User is not on a page related to the 3rd party app


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
Back
User experience best practices
User experience best practices will vary depending on the characteristics of your scenario. Follow the questionnaire below to find your best practices.
You selected
A user clicks an element or performs an action that is unable to go through because of an outage
Where is the user when the error occurs?
We are unable to load the next page after the user clicks something
The user is creating or updating a form
The user is updating a settings page
The user is tracking the progress of an operation with a progress bar component
The user is attempting to install an app
The user is attempting to download a file or other information from an app


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
Back
User experience best practices
User experience best practices will vary depending on the characteristics of your scenario. Follow the questionnaire below to find your best practices.
You selected
A user clicks an element or performs an action that is unable to go through because of an outage
The user is creating or updating a form
What is the user doing in the form when the error occurs?
The user is filling out the form and an outage affects a specific field on the form
The user clicks a submit button at the end of the form and an outage prevents the action from completing


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
Back
User experience best practices
User experience best practices will vary depending on the characteristics of your scenario. Follow the questionnaire below to find your best practices.
You selected
A user clicks an element or performs an action that is unable to go through because of an outage
The user is creating or updating a form
The user is filling out the form and an outage affects a specific field on the form
Is the field required?
The impacted field is a required field on the form
The impacted field is NOT a required field on the form


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
Back
UX best practices for your scenario
The best practices outlined below will be applicable to most use cases associated with the scenario below. However, you may need to combine best practices if your error affects multiple user scenarios, or you might modify the components and content below to best fit your unique use case.
Remember to consult the Engineering best practices in tandem with these recommendations.
You selected
A user clicks an element or performs an action that is unable to go through because of an outage
The user is creating or updating a form
The user is filling out the form and an outage affects a specific field on the form
The impacted field is a required field on the form
Overview
In this scenario, the user clicked a create or edit button (or equivalent), and found that there is missing or out-of-date information in one of the fields on the page. Because the field is required, the form can't be created or edited correctly without the information.
Best practices
There are two scenarios to account for here:
1. Information is unable to be displayed
2. Information is displayed, but could be out-of-date

SI Reliability Best Practices | SI Reliability Best Practices - ⭐

Best practices

There are two scenarios to account for here:

1. **Information is unable to be displayed**
2. **Information is displayed, but could be out-of-date**

Scenario 1: Information is unable to be displayed

In this scenario, the user is blocked from submitting the form because information in a required field can not be displayed at all.

Error alert

To prevent the user from getting all the way down the form or page before realizing that something isn't right, we want to inform the user as soon as possible that information is temporarily unavailable. We do this by putting an error alert at the top of the form to tell the user that what they're trying to do won't work right now.

We're having trouble connecting to [App Name] right now.

Until a connection is reestablished, we won't be able to [create/update your object]. This is due to a temporary issue with our connection, and it will most likely need no action on your part to resolve. Please try again later.

DO

✅ Take the blame for the outage and provide users with a helpful next step they can take in the meantime while the outage is resolved.

We're having trouble connecting to QuickBooks right now.

Until a connection is reestablished, we won't be able to create a QuickBooks invoice from HubSpot...

DON'T

❌ Don't blame the user or the 3rd party app. Errors happen, and the only thing we can control is our response to them.

QuickBooks is experiencing an outage.

You can't create a QuickBooks invoice from HubSpot right now...

DO

✅ Follow the content template below.

Summary of problem

We're having trouble connecting to [App Name] right now.

Impact to the user

Until a connection is reestablished, we won't be able to...

Explanation of solution

This is due to a temporary issue with our connection, and it will most likely need no action on your part to resolve.

Next step

Please try again later.

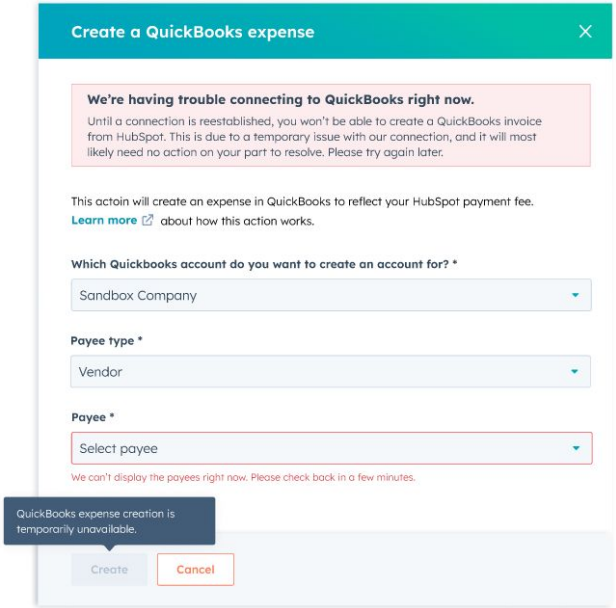
Contextual messaging

The next thing we'll do is provide contextual messaging so the user can find where the error is occurring. Display the field as an error, and add an error message below the field.

Example

QUICKBOOKS
GET - v3/company/<realmId>/query

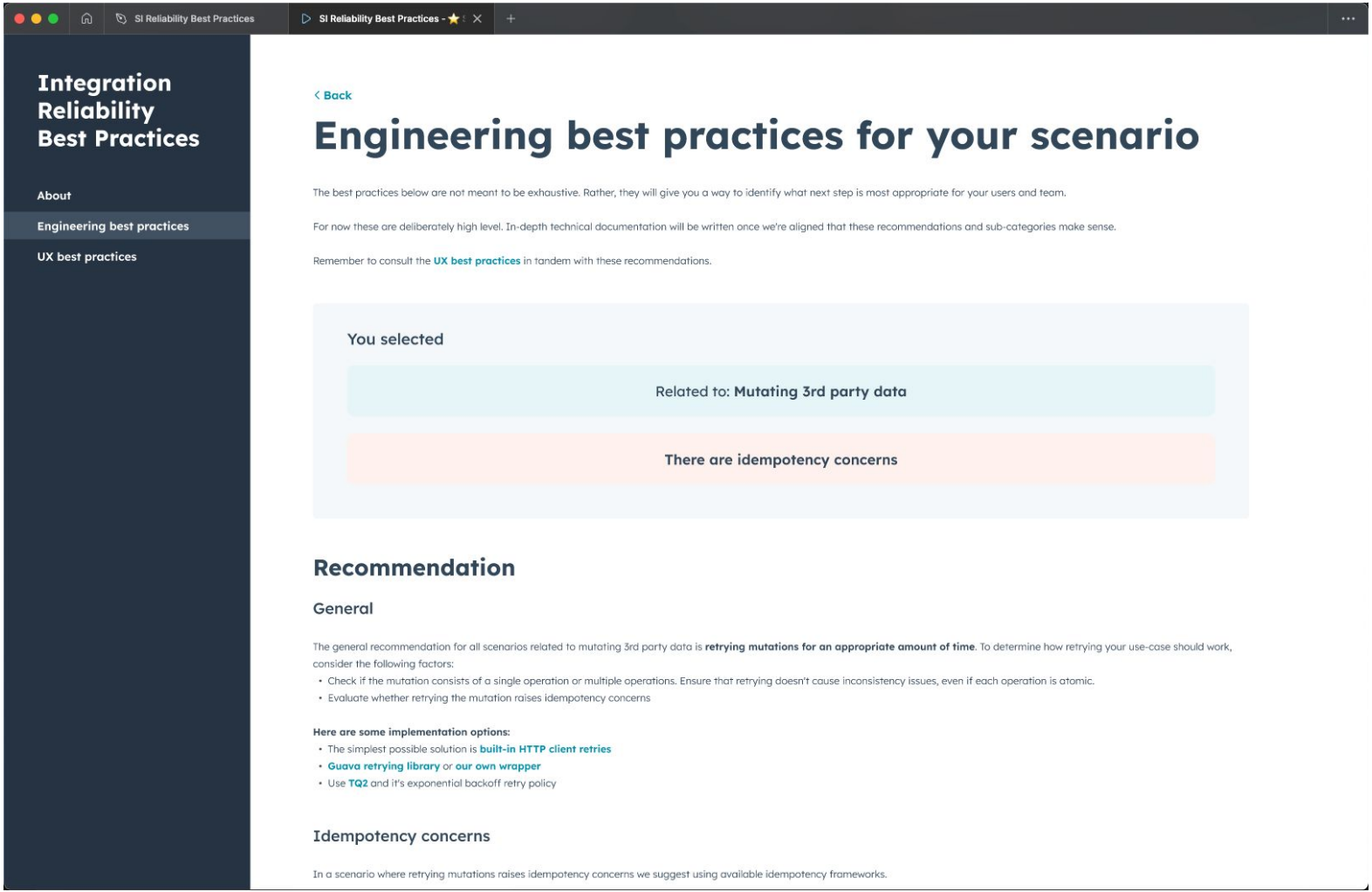
The user wants to create an expense in QuickBooks. Associating the expense with a payee in QuickBooks is a required field on the form, but we can't get a list of their payees to display for the user to choose from. Because this field is required, the user will not be able to submit the form to create the expense until the outage is resolved.



Scenario 2: Information is displayed, but could be out-of-date

In this scenario, the user is able to successfully fill out the form, but information in a required field could be out of date due to the outage.

Warning alert
To prevent the user from getting all the way down the form or page before realizing that something isn't right, we want to inform the user as soon as possible that information is out-of-date. We do this by putting a warning alert at the top of the form to tell the user that what some information might be incorrect or missing.


SI Reliability Best Practices
SI Reliability Best Practices - ⭐
Integration Reliability Best Practices
About
Engineering best practices
UX best practices
Back
Engineering best practices for your scenario
The best practices below are not meant to be exhaustive. Rather, they will give you a way to identify what next step is most appropriate for your users and team.
For now these are deliberately high level. In-depth technical documentation will be written once we're aligned that these recommendations and sub-categories make sense.
Remember to consult the UX best practices in tandem with these recommendations.
You selected
Related to: Mutating 3rd party data
There are idempotency concerns
Recommendation
General
The general recommendation for all scenarios related to mutating 3rd party data is retrying mutations for an appropriate amount of time. To determine how retrying your use-case should work, consider the following factors:
• Check if the mutation consists of a single operation or multiple operations. Ensure that retrying doesn't cause inconsistency issues, even if each operation is atomic.
• Evaluate whether retrying the mutation raises idempotency concerns
Here are some implementation options:
• The simplest possible solution is built-in HTTP client retries
• Guava retrying library or our own wrapper
• Use TQ2 and it's exponential backoff retry policy
Idempotency concerns
In a scenario where retrying mutations raises idempotency concerns we suggest using available idempotency frameworks.

ACTION

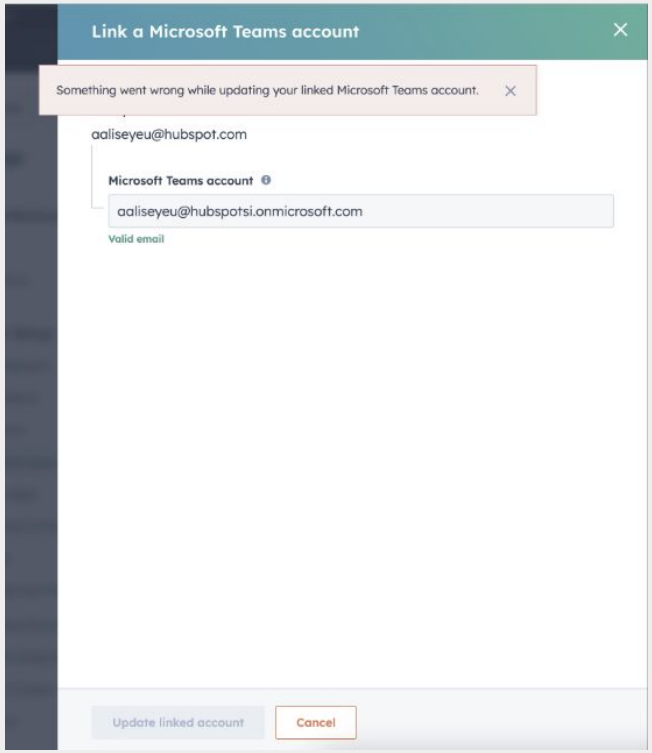
Assign best practices to each relevant API call

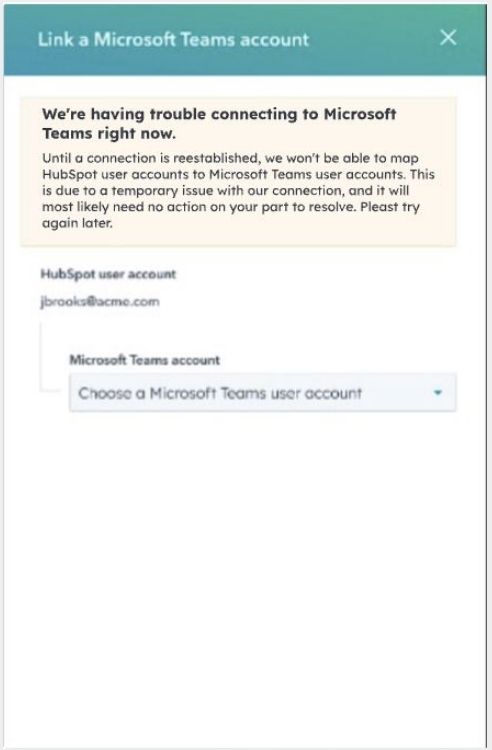
Future UX Responses	Totals	
Full page empty state	18	
Half page empty state	1	
Contextual error alert + messaging	51	
Contextual warning alert + messaging	96	
Warning alert on settings page	93	
Error temporary alert	28	410
Warning temporary alert	1	
Full page modal	5	
Error message (other)	86	
Workflows error	23	
Error in 3rd party app	8	
TBD - unsure of use case	65	65
	475	

Results

RESULTS

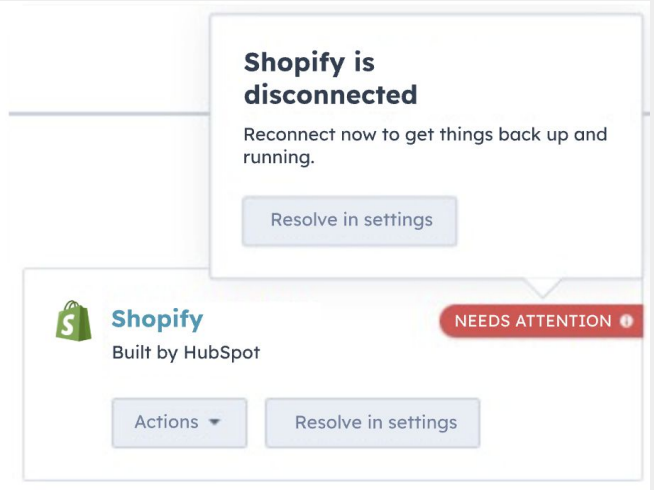
Examples

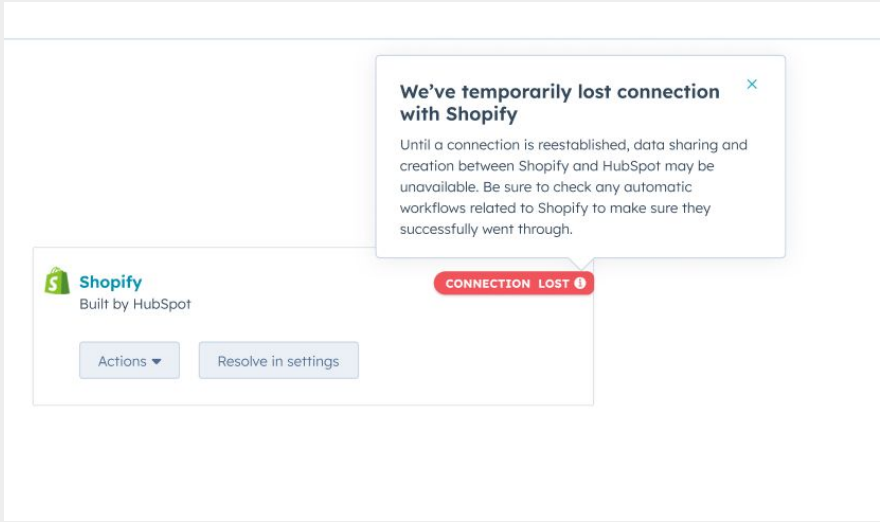




RESULTS

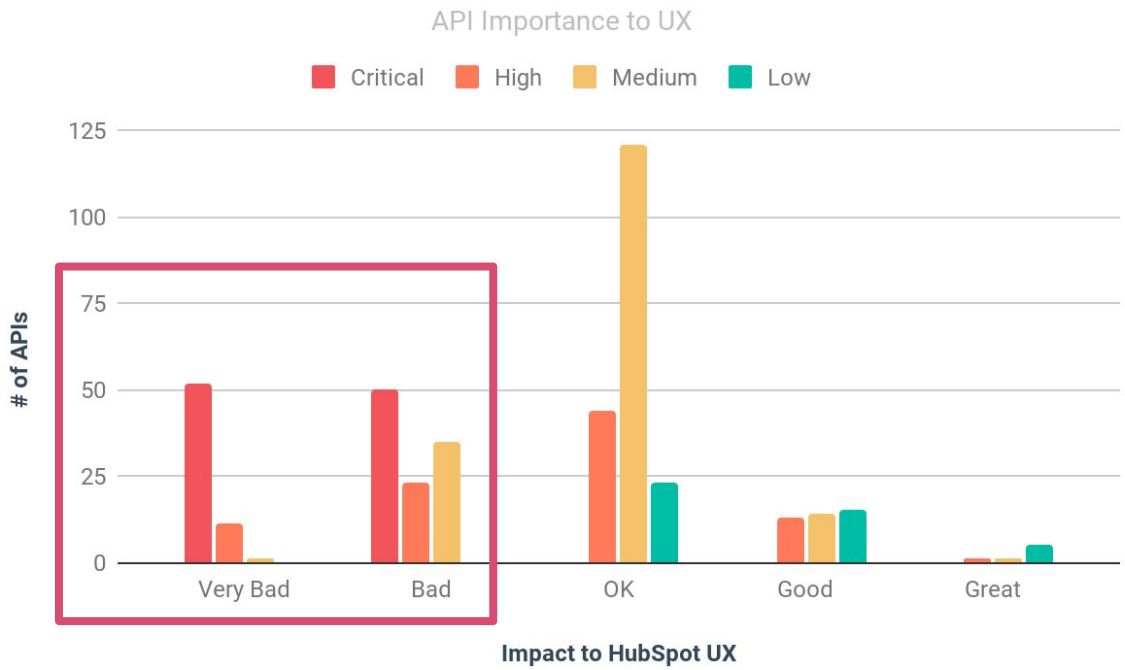
Examples





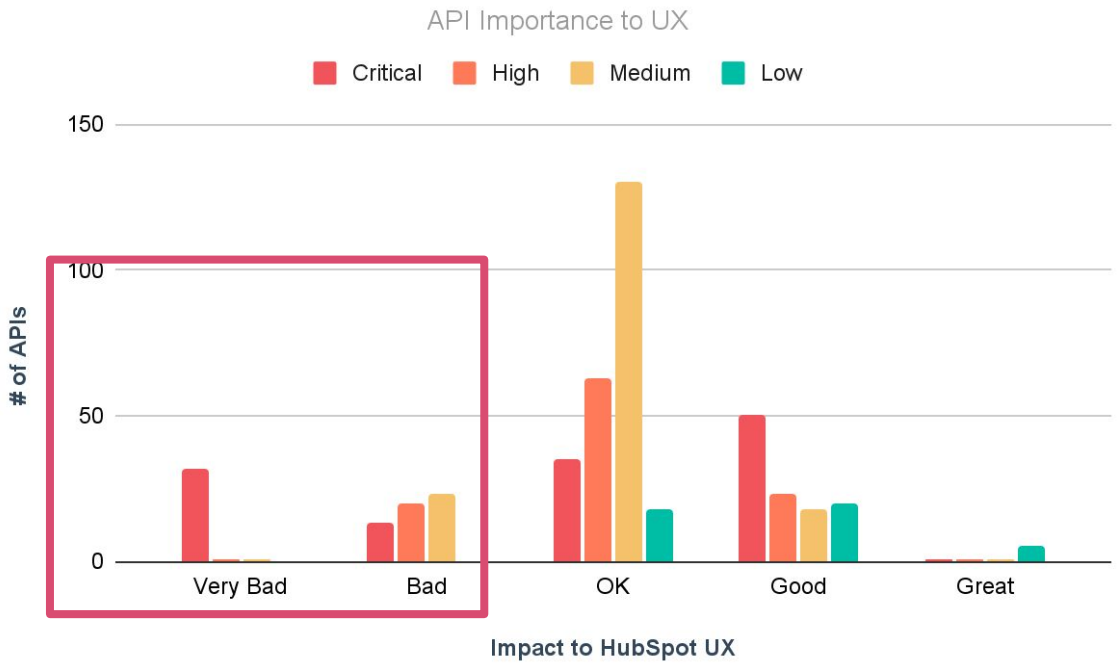
RESULTS

Before: a high amount of critical and very bad impacts to UX.



RESULTS

After: we improved the UX response significantly for our highest impact API calls.



Thank you!